

Remarks

The above Amendments and these Remarks are in reply to the Office Action mailed March 8, 2006. No fee is due for the addition of any new claims.

Claims 1-10, 13, 16-18, and 21-25 were pending in the Application prior to the outstanding Office Action. In the Office Action, the Examiner rejected claims 1-10, 13, 16-18, and 21-25. The present Reply amends claim 13, leaving for the Examiner's present consideration claims 1-10, 13, 16-18, and 21-25. Reconsideration of the rejections is requested.

I. Summary of Examiner's Rejections

Claims 1-10, 13, 16-18, and 21-25 were rejected under 35 U.S.C. §103(a) as being unpatentable over Jones et al. (U.S. Patent No. 6,877,163) in view of Dattke (U.S. Patent Application Publication No. 2004/0143835).

II. Summary of Applicant's Response

The present Response amends Claim 13 to correct minor informalities. No new matter has been added, leaving for the Examiner's present consideration Claims 1-10, 13, 16-18 and 21-25. Reconsideration of the Application, as amended, is respectfully requested. Applicant respectfully reserves the right to prosecute any originally presented or canceled claims in a continuing or future application.

III. Response to Rejections

Claim Objections

In the Office Action, Claim 13 was objected to because of the following informalities: claim 13 depends from cancelled claim 12. Claim 13 has been herein amended to obviate the objection.

Claim Rejections under 35 U.S.C. §103(a)

In the Office Action, Claims 1-10, 13, 16-18, and 21-25 were rejected under 35 U.S.C. 103(a) as being unpatentable over U.S. Patent No. 6,877,163 to Jones et al. (hereinafter Jones) in view of U.S. Patent Application Publication No. 2004/0143835 to Dattke et al. (hereinafter Dattke).

Claim 1

The Office Action agreed that Jones does not teach a wrapper class comprising at least one of vendor specific extension methods from the vendor class. The Office Action asserted, however, that in order to provide “customers of the standard application to customize the features of the standard application by providing customer-specific extensions for the features implemented by the standard application [p. 1, paragraph 0002 of Dattke] and a dynamic proxy class can be used to create a type-safe proxy object for a list of interfaces without requiring pre-generation of the class prior to compilation [p. 1, paragraph 0005 of Dattke],” that “it would have been obvious to a person of ordinary skill in the art at the time the invention was made to apply the teaching of a wrapper class comprising at least one of vendor specific extension methods from the vendor class as taught by Dattke to the invention of Jones.”

Applicants respectfully traverse.

In view of the present invention, Applicant fully agrees with the Examiner that embodiments of the invention may indeed provide for overlaying an “extension factory” architecture for Business AddIns (BadI) for SAP as taught by Dattke (Dattke, [0003], [0031]) onto classes retrieved from a vendor object, or modifying the Jones reference to convert classes specified in a list of interfaces specified at runtime (Jones, Abstract) into extension properties for inclusion into Dattke’s extension registry (Dattke, [0031]) instead of generating a proxy class 202 and a proxy class instance 204 (Jones, col. 5, lines 50 – 62) for classes specified in the list of interfaces specified at runtime, however, NEITHER ONE of such overlaying or modifying is taught in the asserted Jones/Dattke combination. Because neither Jones nor Dattke, alone or in any combination teach, suggest or

otherwise render obvious such mechanism for generating a wrapper class comprising at least one of vendor specific extension methods from the vendor class (instead of the extension factory structure presently implemented), and because modifications to Dattke to do so would change Dattke' principle of operation in a manner contrary to their stated purpose, **the idea to do so must be drawn via impermissible hindsight from the present application.**

Jones teaches "an object-oriented data processing system ... that provide[s] a proxy class dynamically generated at runtime that implements a list of interfaces specified at runtime such that a method invocation through an interface on an instance of the class is encoded and dispatched uniformly to an object that performs the invocation of the requested method." (Abstract). Not only does Jones require a list of interfaces to be specified at runtime, but, as noted by the Office Action, Jones does not teach "a wrapper class comprising at least one of vendor specific extension methods from the vendor class."

Dattke state that their work is directed to solving the problems inherent to using SAP's BAdI interface for incorporating user extensions into SAP's programming interfaces:

In order to extend the standard SAP application, a Business Add-In (BAdI) is defined by the application developer for the standard application. The application developer also defines an interface for the BAdI. The defined interface for the BAdI is used to create an adapter class for implementing the BAdI. The adapter class is used by the extension developer developing the application extension to provide an implementation of the BAdI. The application developer extends the standard application by creating an instance of the adapter class in the standard application and calling the corresponding methods of the adapter class at the appropriate time. In order to use a BAdI, an extension developer must provide his own implementation of the BAdI by implementing the enhancements defined by the BAdI and activating the implementations of the enhancements. (Dattke, [0003])

Dattke's approach to solving this problem uses an extension factory that "receive[s] an application call to an extension method defined in an extension object definition, the extension object definition having associated extension object implementations, the extension object implementations providing extension method implementations of the extension method; obtain results by calling multiple extension method implementations of the extension method; and respond to the application call using the results obtained from the extension method implementations." (Dattke, Abstract; claim 1) [emphasis added].

Because Dattke's extension factory is intended to provide a user with the capability to incorporate defined user extension objects into SAP's programming interfaces (Dattke, [0003]), modifying Dattke's system to include techniques that receive classes from Jones system; (somehow) derive "extension properties, e.g., BAdI properties and filter values" (Dattke, FIG. 2: box 105; [0031]); and then incorporate this information into their registry in order to allow producing a wrapper class comprising the vendor class (requires, extra time, and extra steps) would require modifications to Dattke's purpose as well as Dattke's principle of operation (see MPEP § 2143.01) because such modifications would NECESSARILY burden Dattke's system contrary to their stated purpose: incorporating user extension objects into SAP's programming interfaces.

Therefore, the idea to so modify either one or both of Dattke or Jones must be drawn via impermissible hindsight from the present application.

For at least this reason, the rejection is improper and should be withdrawn.

Claims 10, 21, 22 and 24

Claims 10, 21, 22 and 24 are not addressed separately, but it is respectfully submitted that these claims have substantially similar features as those discussed above regarding claim 1. Applicant respectfully submits that Claims 10, 21, 22 and 24 are similarly neither anticipated by, nor rendered obvious in view of the cited references, and reconsideration thereof is respectfully requested.

Claims 2-9, 13, 16-18, 23 and 25

Claims 2-9, 13, 16-18, 23 and 25 are not addressed separately, but it is respectfully submitted that these claims are allowable as depending from an allowable independent claim, and further in view of the comments provided above. Applicant respectfully submits that Claims 2-9, 13, 16-18, 23 and 25 are similarly neither anticipated by, nor rendered obvious in view of the cited references, and reconsideration thereof is respectfully requested.


It is also submitted that these claims also add their own limitations which render them patentable in their own right. Applicant respectfully reserves the right to argue these limitations should it become necessary in the future.

In light of the above, it is respectfully submitted that all of the claims now pending in the subject patent application should be allowable, and a Notice of Allowance is requested. The Examiner is respectfully requested to telephone the undersigned before an advisory action is issued in order to avoid any unnecessary filing of an appeal.

The Commissioner is authorized to charge any underpayment or credit any overpayment to Deposit Account No. 06-1325 for any matter in connection with this response, including any fee for extension of time, which may be required.

Respectfully submitted,

Date: MAY 8, 2006

By: 
Paul A. Durdik
Reg. No. 37,819

Customer No. 23910
FLIESLER MEYER LLP
Four Embarcadero Center, Fourth Floor
San Francisco, California 94111-4156
Telephone: (415) 362-3800 x227
Facsimile: (415) 362-2928